

# Application of single neuron adaptive PID controller during the process of timber drying

ZHANG Dong-yan<sup>1</sup>, LIU Ya-qiu<sup>2</sup>, CAO Jun<sup>1</sup>

<sup>1</sup> Northeast Forestry University, Harbin 150040, P. R. China

<sup>2</sup> Harbin Industry University, Harbin 150001, P. R. China

**Abstract:** The paper presents a method of using single neuron adaptive PID control for adjusting system or servo system to implement timber drying process control, which combines the thought of parameter adaptive PID control and the character of neural network on exactly describing nonlinear and uncertainty dynamic process organically. The method implements functions of adaptive and self-learning by adjusting weighting parameters. Adaptive neural network can make some output trail given hoping value to decouple in static state. The simulation result indicates the validity, veracity and robustness of the method used in the timber drying process

**Keywords:** Process control; Timber drying; Single neuron; Adaptive control; PID control

**CLC number:** S781.71

**Document code:** A

**Article ID:** 1007-662X(2003)03-0244-05

## Introduction

The process of timber drying is different from other drying process. It has particular complexity according to its chemical and physical characteristics. It is an urgently concerned problem to improve control level of drying process for timber drying industry, which aims at implementing full automatic control on indeed meaning. Whether the parameters of drying media can be real-time controlled automatically for drying process has become main influencing factor to the quality of timber drying. Timber drying kiln is a complicated system with temperature and humidity coupling, meanwhile it is also a big inertia, big time lag and non-linear system. How to control it better, the experts both domestic and abroad are making many tries about it.

## Control system

Based on adaptive neural network, we designed a control system, which can real time control the parameters of drying media for timber drying process. According to the requirement of timber drying technology, the control variables of the system are temperature, humidity and wind speed of drying media.

The core of principle for single neuron adaptive PID control is based on PID control, adaptive and self-learning of single neuron. The optimizing idea mainly embodies learning rule of neural network to form different control algorithms, and uses these algorithms such as supervising

Hebb learning algorithm, the algorithm of making error square as performance index and so on to modify weight value for determining an optimizing control sequence. It is a control method with stronger robustness and advantageous to conquer time lag and time varying. As an identifying model, the neural network is a physical implement for actual system. The algorithm of single neuron is simple and may be used in on-line control.

Single neuron adaptive PID control system for timber drying kiln is composed of timber drying kiln and neural network controller. Its principle is shown as Fig.1, in which  $x_j^i(t)$  is No.  $j$  input state for No.  $i$  neuron;  $w_j^i$  weight value of corresponding  $x_j^i(t)$ ;  $p_i(t)$  and  $K_i$  is descending signal and proportion coefficient, respectively. The controller form is setting a adaptive neuron on each input-output main chain to get control signal  $u_i(t)(i=1,2)$ . The input value of converter reflects the state variables such as controlled object and control index. If there are given value  $r_i(t)$ , output measuring value  $y_i(t)(i=1,2)$ , they will be converted to state variable  $x_j^i(t)(i=1,2; j=1,2,3)$  for the need of adaptive neural network learning control.

Adaptive neural network can make certain output trail given hoping value to decouple in static state. Therefore, there is a single neuron adaptive PID controller on input-output main chain of temperature and humidity during timber drying process respectively, which implements decoupling control.

## Single neuron adaptive PID controller

### Adaptive neuron and learning algorithm

Standardization learning algorithm can be used to ensure astringency and control robustness, whose controller is

**Foundation item:** This study is supported by the Key Technologies R&D Program of Harbin (0111211102).

**Biography:** ZHANG Dong-yan (1976- ), female, Ph.D. candidate, assistant professor in Northeast Forestry University, Harbin 150040, P. R. China  
Email: Zhang\_dongyan@yahoo.com.cn

**Received date:** 2003-05-23

**Responsible editor:** Song Funan

$$u(t) = K \sum_{i=1}^n w'_i(t) x_i(t) \quad (1)$$

$$w'_i(t) = w_i(t) / \sum_{i=1}^n |w'_i(t)| \quad (2)$$

$$w_i(t+1) = w_i(t) + d[r(t) - y(t)]x_i(t) \quad (3)$$

Where,  $r(t)$  is given signal;  $y(t)$  is actual output of object.

Before using above all to form loop-locked controller, confirming the quantity ( $n$ ) of state variables ( $x_i$ ) and property of  $x_i(t)$  is a requisite. Because timber drying kiln is adjusting system, namely when input  $r(t)$  is certain fixed-value signal,  $n$  may be 3, and state variable  $x_i(t)$  is:

$$x_1(t) = r(t)$$

$$x_2(t) = r(t) - y(t) = e(t)$$

$$x_3(t) = \Delta x_2(t) = x_2(t) - x_2(t-1)$$

The formed control system is shown as Fig.2.

According to Fig.2 the control signals produced by neural network are composed of three sections: feed-forward proportional control  $u_1(t)$ , feedback proportional control  $u_2(t)$ , and feedback differential control  $u_3(t)$ . It is a kind of multi-layer and multi-mode control structure, gathering feed-forward and feedback as a whole, which are correlative and compensative mutually. In feed-forward control, given signal  $r(t)$  directly acts on controlled object through  $w'_1(t)$ , which quickens response speed of system. Feedback proportional control may diminish trailing error quickly, while feedback differential control not only can improve response speed but diminish overshooting value as well. The weight value  $w'_i(t)$  ( $i=1, 2, 3$ ) reflects dynamic character of controlled object and process response, and neuron adjusts  $w'_i(t)$  ( $i=1, 2, 3$ ) continually by its learning strategy. As a result, relevancy action of three controls can eliminate deflection quickly, and then ensure system be in stable state (Li 2001; Shu 1995).

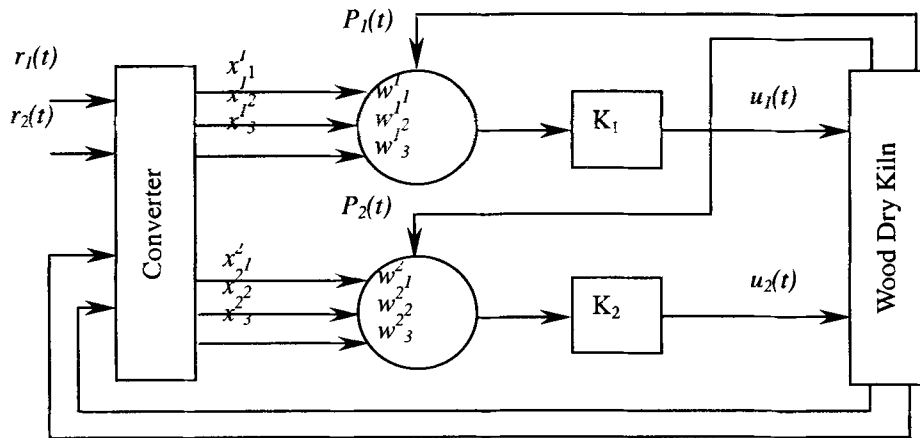


Fig.1 Temperature and humidity adaptive neural network control system

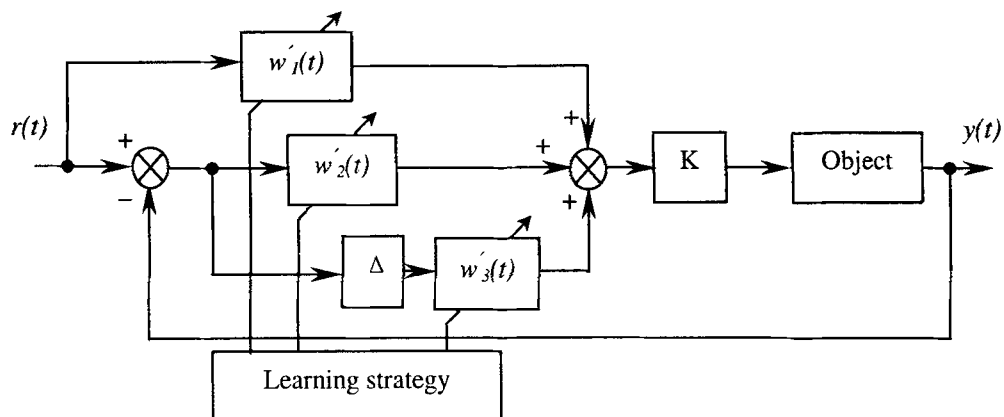


Fig.2 Single neuron control system

### Single neuron PID controller and its learning algorithm

Structure diagram is shown as Fig.3.

In which the input is given value  $r(k)$  and the output value is  $y(k)$  for converter. The outputs are the needed state variable  $x_1$ ,  $x_2$  and  $x_3$  for neuron control.

$$\left. \begin{aligned} x_1(k) &= r(k) - y(k) \\ x_2(k) &= \Delta e(k) \\ x_3(k) &= e(k) - 2e(k-1) + e(k-2) \end{aligned} \right\} \quad (4)$$

Here  $z(k)=x_1(k)=e(k)$  is performance index or recursive signal. As it shows,  $K$  is proportional coefficient of neuron,

and  $k>0$ . Neuron produces control signal by relevancy search, namely,

$$u(k) = u(k-1) + K \sum_{i=1}^3 w_i(k) x_i(k) \quad (5)$$

where,  $w_i(k)$  is weight coefficient corresponding to  $x_i(k)$ . Single neuron adaptive PID controller implements the function of adaptive and self-learning just by adjusting weight coefficients, which can form different control algorithms using different learning rules.

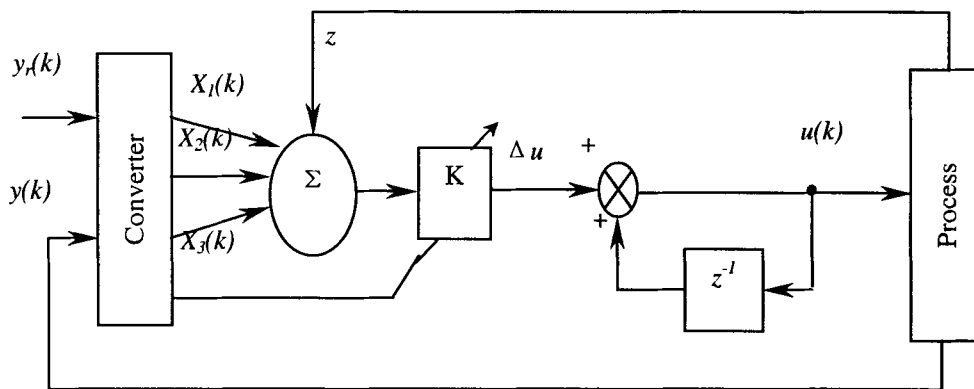


Fig.3 Single neuron PID controller structure

#### (1) Single neuron adaptive PID controller by supervising Hebb learning algorithm

Considering weight coefficient  $w_i(k)$  should be concerned with the correlative functions of input, output and output error of neuron, while using supervising Hebb learning algorithm, there are

$$w_i(k+1) = (1-c)w_i(k) + \eta v_i(k) \quad (6)$$

$$v_i(k) = z(k)u(k)x_i(k) \quad (7)$$

where,  $v_i(k)$  is recursive signal and hushes gradually during the process;  $z(k)$  is output error signal, and  $z(k)=r(k)-y(k)=e(k)$ ;  $\eta$  is learning rate, and  $\eta>0$ ;  $c$  is constant, and  $0 \leq c < 1$ .

Substituting equation (7) into equation (6),

$$\Delta w_i(k) = w_i(k+1) - w_i(k) = -c[w_i(k) - \frac{\eta}{c} z(k)u(k)x_i(k)] \quad (8)$$

If function  $f_i(w_i(k), z(k), u(k), x_i(k))$  is exiting, then to calculate the partial differential of  $w_i(k)$  there is

$$\frac{\partial f_i}{\partial w_i} = w_i(k) - \frac{\eta}{c} g_i(z(k)u(k)x_i(k))$$

So equation (8) may be written as

$$\Delta w_i(k) = -c \frac{\partial f_i(\bullet)}{\partial w_i}$$

According to the equation above, the amending of weight coefficient is realized along the minus gradient direction of  $w_i(k)$  corresponding to function  $f_i(\bullet)$ . With stochastic approach theory, We can prove that  $w_i(k)$  can constringe to a certain stable value  $w_i^*$  when  $c$  is small enough, and the deflection between  $w_i(k)$  and its hoping value is in allowable range.

To ensure the astringency and robustness of equation (8) and  $u(k)$  for learning algorithm of single neuron adaptive PID control, learning algorithms should be disposed canonically, then

$$\left. \begin{aligned} u(k) &= u(k-1) + K \sum_{i=1}^3 \bar{w}_i(k) x_i(k) \\ \bar{w}_i(k) &= \frac{w_i(k)}{\sum_{i=1}^3 |w_i(k)|} \\ w_1(k+1) &= w_1(k) + \eta_i K z(k+1) u(k) x_1(k) \\ w_2(k+1) &= w_2(k) + \eta_p K z(k+1) u(k) x_2(k) \\ w_3(k+1) &= w_3(k) + \eta_D K z(k+1) u(k) x_3(k) \end{aligned} \right\} \quad (9)$$

Where,  $\eta_i$ ,  $\eta_p$  and  $\eta_D$  is proportional learning rate, integral and differential learning rate respectively. In the other word, different learning rate is separately used in proportional (P), integral (I) and differential (D) control, so it is easy to adjust weight coefficient respectively according to the need. And their values may be confirmed in advance by spot experiment or simulating research, meanwhile let  $c=0$ .

(2) *Single neuron adaptive PID controller by algorithm of making error square as performance index*

Firstly, importing performance index function is as follows:

$$J = \frac{1}{2} [r(k+1) - y(k+1)]^2 = \frac{1}{2} z^2(k+1)$$

It is to make the amending for weight coefficient  $w_i(k)$  along the minus direction of  $J$ , which will have more specific physical meaning to searching adjust for minus direction of  $w_i(k)$ . The gradient of  $J$  concerning  $w_i(k)$  is

$$\frac{\partial J}{\partial w_i(k)} = -z(k+1) \frac{\partial y(k+1)}{\partial u(k)} \cdot \frac{\partial u(k)}{\partial w_i(k)}$$

The adjusting value of  $w_i(k)$  is

$$\begin{aligned} \Delta w_i(k) &= w_i(k+1) - w_i(k) = -\eta_i \frac{\partial J}{\partial w_i(k)} \\ &= \eta_i z(k+1) \frac{\partial y(k+1)}{\partial u(k)} \cdot \frac{\partial u(k)}{\partial w_i(k)} \end{aligned} \quad (10)$$

Using equation (4) and substituting (5) into (10) we can get

$$\left. \begin{aligned} \Delta w_1(k) &= \eta_i K z(k+1) e(k) \frac{\partial y(k+1)}{\partial u(k)} \\ \Delta w_2(k) &= \eta_p K z(k+1) [e(k) - e(k-1)] \frac{\partial y(k+1)}{\partial u(k)} \\ \Delta w_3(k) &= \eta_D K z(k+1) [e(k) - 2e(k-1) + e(k-2)] \frac{\partial y(k+1)}{\partial u(k)} \end{aligned} \right\} \quad (11)$$

$\partial y(k+1)/\partial u(k)$  is often unknown in PID control algorithm, and it may be replaced approximately with symbol function  $\text{sgn}[\partial y(k+1)/\partial u(k)]$ . The effect of loose calculating from it may be compensated by adjusting learning rate  $\eta_i$ . After canonically settling above algorithms we can get:

$$\left. \begin{aligned} u(k) &= u(k-1) + K \sum_{i=1}^3 w_i(k) x_i(k) \\ \bar{w}_i(k) &= \frac{w_i(k)}{\sum_{i=1}^3 |w_i(k)|} \\ w_1(k+1) &= w_1(k) + \eta_i K z(k+1) x_1(k) \text{sgn} \left[ \frac{\partial y(k+1)}{\partial u(k)} \right] \\ w_2(k+1) &= w_2(k) + \eta_p K z(k+1) x_2(k) \text{sgn} \left[ \frac{\partial y(k+1)}{\partial u(k)} \right] \\ w_3(k+1) &= w_3(k) + \eta_D K z(k+1) x_3(k) \text{sgn} \left[ \frac{\partial y(k+1)}{\partial u(k)} \right] \\ \text{sgn}(x) &= \begin{cases} +1 & x \geq 0 \\ -1 & x < 0 \end{cases} \end{aligned} \right\} \quad (12)$$

$x_1(k)$ ,  $x_2(k)$  and  $x_3(k)$  is similar to equation (4).

## Simulation research

It is very difficult to get premise mathematical model for temperature and humidity coupling complicated system which is a big inertia, big time lag and nonlinear like timber drying kiln. We use a small drying kiln (its capacity is 0.5 m<sup>3</sup>) for experiment. It is electronic heating and cold spray. Supposed the kiln is a double input and outputs coupling system, we can get approximate linear model ( $G_1(s)$ ) on certain working point by local linearization after fitting its data of step response to first order inertia model with pure lag tache.

$$G_1(s) = \begin{bmatrix} \frac{2.58 e^{-128 s}}{908 s + 1} & \approx 0 \\ -\frac{0.12 e^{-30 s}}{102 s + 1} & \frac{1.8 e^{-35 s}}{86 s + 1} \end{bmatrix}$$

In actual testing, parameters change along with the factors such as working point, disturbance and so on. Modifying corresponding state variable on-line is the embodiment of the adaptability to control object of single neuron adaptive PID controller and can get ideal response curve in bigger range. Fig.4(a) is step response of  $G_1(s)$ . If the function affected by environment becomes  $G_2(s)$ ,

$$G_2(s) = \begin{bmatrix} \frac{2.58 e^{-300 s}}{1500 s + 1} & \approx 0 \\ -\frac{0.12 e^{-100 s}}{200 s + 1} & \frac{1.8 e^{-100 s}}{200 s + 1} \end{bmatrix}$$

Which also can keep system stable but not modify any parameter. The step response curve of  $G_2(s)$  is shown as Fig.4(b).

## Conclusion

Single neuron adaptive PID control can implement decoupling automatically for coupling system that has better

adaptive capability with the algorithm characters. The testing result of actual application for drying kiln indicated that

the method had better characteristic of real-time and robustness for being used in timber drying process control.

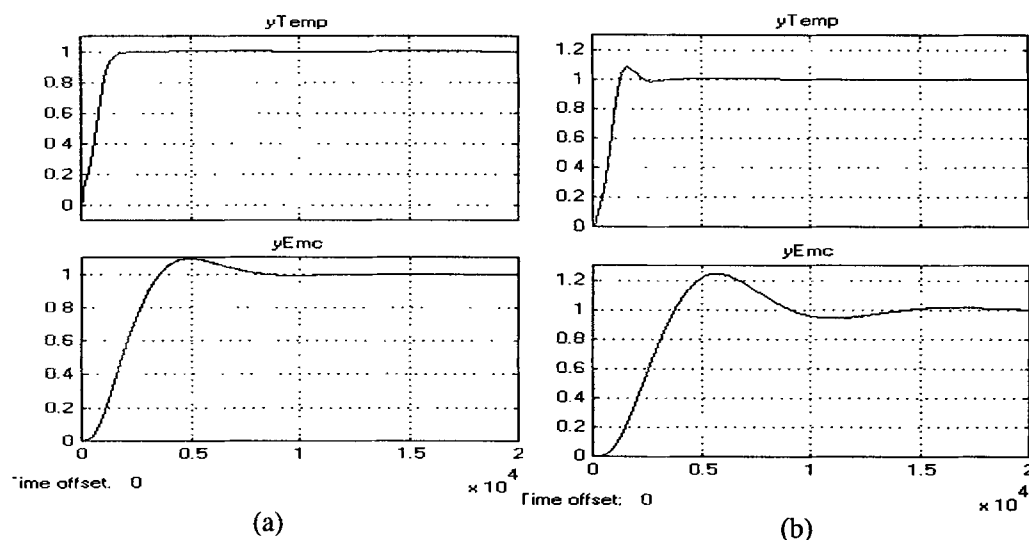


Fig.4 (a) Step response curve of temperature and humidity for drying kiln  
(b) Step response curve after parameter of model changing

## References

- Li Yuyun. 2001. Neural network predicting controller for non-linear heating furnace [J] China Mechanical Engineering, 2: 216-219.
- Shu Diqian. 1995. Single neuron adaptive PID controller and its application in electric heating kiln [J]. Electric Drive, 25(1): 29-32.
- Tjokro, S., Shah, S.L. 1985. Adaptive PID Control [C]. Proceedings of the 1985 America Control Conference, 1528-2392.
- Wang Ning. 1993. Adaptive neural network control for multi-variable system [J]. Pattern Recognition and Artificial Intelligence, 6(3): 247-252.